

**DESARROLLO DE UN ENTORNO VIRTUAL PARA VISUALIZACIÓN 3D DE
LOS RIESGOS QUE SE PUEDEN PRESENTAR DENTRO DE LAS
INSTALACIONES DE LA UNIVERSIDAD AUTÓNOMA DE OCCIDENTE**

LINA JOHANNA MONTOYA AGUILAR

**UNIVERSIDAD AUTÓNOMA DE OCCIDENTE
FACULTAD DE INGENIERIA
DEPARTAMENTO DE OPERACIONES Y SISTEMAS
PROGRAMA INGENIERÍA INFORMÁTICA
SANTIAGO DE CALI
2011**

**DESARROLLO DE UN ENTORNO VIRTUAL PARA VISUALIZACIÓN 3D DE
LOS RIESGOS QUE SE PUEDEN PRESENTAR DENTRO DE LAS
INSTALACIONES DE LA UNIVERSIDAD AUTÓNOMA DE OCCIDENTE**

LINA JOHANNA MONTOYA AGUILAR

Trabajo de Grado para optar al título de Ingeniero en Informático

**Director
JESÚS DAVID CARDONA QUIROZ
Doctor en Ingeniería Informática**

**UNIVERSIDAD AUTÓNOMA DE OCCIDENTE
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE OPERACIONES Y SISTEMAS
PROGRAMA INGENIERÍA INFORMÁTICA
SANTIAGO DE CALI
2011**

Nota de aceptación:

**Aprobado por el Comité de Grado
en cumplimiento de los requisitos
exigidos por la Universidad
Autónoma de Occidente para optar
al título de Ingeniero en
Informática.**

Ms.C. CARLOS ALBERTO PELÁEZ
Jurado

Ph.D. OLMEDO ARCILA GUZMÁN
Jurado

Santiago de Cali, 01 de Junio de 2011

Primero a DIOS, por darme la bendición de tener Educación Superior gracias a Él que me dio sabiduría y entendimiento, que no me deja rendir y que siempre me acompaña en todo momento.

A mis Padres, Vilma Aguilar y Freddy Montoya, quienes más que la vida, me han brindado los mejores consejos, el apoyo cuando lo necesito, por todo lo que soy; gracias a su esfuerzo me ayudan cada día a crecer, a soñar, a tomar decisiones, a ser mejor. Agradezco en especial a mi madre que siempre ha estado ahí conmigo, a quien debo mi dedicación, ánimos de lucha y esfuerzo.

A mi hermana, Ximena Montoya, por el apoyo constante, por darme las fuerzas y energías cuando las necesito, por escucharme y comprenderme, por estar ahí siempre.

A mi abuelita, Blanca Martínez, por alentarme cada día, por hacerme saber que siempre he contado y contaré con ella en las decisiones que tome en mi vida. A Orlando Martínez que siempre me ha apoyado en las decisiones que he tomado, por estar presente en cada momento y por alentarme cada día a crecer. A mis primos, que con sus conversaciones y bromas, me han hecho sentir una persona muy especial y muy feliz de ser parte de la misma familia.

AGRADECIMIENTOS

A los Profesores de la Universidad Autónoma de Occidente por estar presentes siempre que lo necesité, en especial a mi director Jesús David Cardona, por creer en mí y por permitirme ser parte de un proyecto como éste.

A la directora de Ingeniería Informática: Lyda Peña por orientarme durante toda mi carrera.

A la Universidad por brindarme tan maravillosas oportunidades y el honor de ser una representante más de esta extraordinaria institución.

A Gisler Garcés que me asesoró con ideas durante todo el proceso de implementación y mejoramiento de esta plataforma.

A mis compañeros autónomos que compartieron conmigo mucho más que conocimiento en las aulas de clase.

A todos ellos Gracias.

CONTENIDO

	pág.
INTRODUCCIÓN	15
1. PLANTEAMIENTO DEL PROBLEMA	17
2. JUSTIFICACIÓN	19
3. ANTECEDENTES	20
4. MARCO TEÓRICO	22
4.1 GRÁFICOS 3D	23
4.2 INTERACCIÓN 3D	28
4.3 REALIDAD VIRTUAL Y ENTORNO VIRTUAL	29
4.4 PLATAFORMAS PARA EL DESARROLLO DE ENTORNOS VIRTUALES	34
4.5 MODELADO Y TEXTURIZADO DE OBJETOS 3D	42
5. OBJETIVOS	47
5.1 OBJETIVO GENERAL	47
5.2 OBJETIVOS ESPECÍFICOS	47
6. METODOLOGÍA	48
7. SINTESIS DE DESARROLLO	50
7.1 METODOLOGÍA DE INGENIERÍA DE SOFTWARE PARA EVS	52
7.2 LENGUAJE DE PROGRAMACIÓN	55
7.3 ARQUITECTURA	56
7.4 COMPONENTES	57
7.5 ETAPA DE IMPLEMENTACIÓN	59
7.5.1 VIRTUAL CAMPUS 3D	59

7.5.2	JAVA WEB START	59
7.5.3	JMONKEY ENGINE	60
7.5.4	MODELOS 3D	60
7.5.5	AVATARES	61
7.5.6	ANIMACIONES	63
7.5.7	TERRENO Y DESPLAZAMIENTO	63
7.6	MÓDULO ADMINISTRATIVO	66
7.6.1	POOL DE RECURSOS	66
7.6.2	ENTORNO WEB	68
7.7	MÓDULO CLIENTE	71
7.7.1	VIRTUAL CAMPUS SCRIPT CREATOR	71
7.7.2	VIRTUAL CAMPUS RISKS	76
7.8	PRUEBAS	83
8	CONCLUSIONES	85
9	RECOMENDACIONES	87
	BIBLIOGRAFÍA	88

LISTA DE TABLAS

	pág.
Tabla 1. Elementos utilizados en los cuatro tipos más importantes de realidad virtual.	34
Tabla 2. Interfaces gráficas de usuario open source	77

LISTA DE FIGURAS

	pág.
Figura 1. Flujo de trabajo de una interacción 3D	29
Figura 2. Arquitectura de un motor de alto nivel	38
Figura 3. Síntesis de la solución para Virtual Campus 3D- Risks	51
Figura 4. Dependencia entre componentes	58
Figura 5. Diagrama de despliegue	58
Figura 6. Geometría 3D que representa ruta de evacuación creado en Wings	61
Figura 7. Exportando avatar a MD5	62
Figura 8. Animaciones importadas a Virtual Campus 3D- Risks con formato MD5	63
Figura 9. Terreno de pruebas para Virtual Campus 3D	64
Figura 10. Terreno inspirado en las instalaciones de la Universidad creado en Blender	65
Figura 11. Modelo 3D final de la Universidad, para Virtual Campus 3D- Risks.	65
Figura 12. Jar de plugins en el pool de recursos de Virtual Campus 3D- Risks	66
Figura 13. Contenido del paquete plugins	67
Figura 14. Módulo administrativo, zona de Plugins	69
Figura 15. Modulo administrativo, estados de aplicación	70
Figura 16. Ubicación del avatar guía	72
Figura 17. Rotación del avatar guía	73

Figura 18. Cuadro de diálogo	73
Figura 19. Subir una imagen como mapa	74
Figura 20. Seleccionar una animación	74
Figura 21. Archivos plano y .XML generados por usuario en Virtual Campus- Script Creator	75
Figura 22. Movimientos de desplazamiento del Avatar Guía	76
Figura 23. Introducción y breve explicación de la barra de menú	76
Figura 24. Cuadro de diálogo con JTextArea	78
Figura 25. Vista de cámara vuelo de pájaro, muestra el Avatar Guía y la ubicación del usuario	79
Figura 26. Partículas de fuego en un objeto en Virtual Campus 3D-Risks	79
Figura 27. Zonas de refugio y rutas de evacuación en 3D	80
Figura 28. Rutas de evacuación, idea tomada del storyboard y la implementada para Virtual Campus 3D- Risks	80
Figura 29. Archivos .XML para importación de objetos de escenario Virtual Campus 3D- Risks	81
Figura 30. Cuadros de diálogo, idea tomada del storyboard y la implementada para Virtual Campus 3D- Risks	83

LISTA DE ANEXOS

La siguiente lista nombra los anexos se encuentran en formato digital disponibles en CD adjunto.

Anexo A. Modelo 3D del entorno virtual

Anexo B. Objetos 3D existentes

Anexo C. Arquitectura del entorno virtual

Anexo D. Prototipo de interfaz gráfica de usuario

Anexo E. Componentes del entorno virtual

Anexo F. Casos de pruebas

Anexo G. Descripción del software

Anexo H. Pruebas de usabilidad

Anexo I. Clasificador del entorno virtual

Anexo J. Requisitos funcionales y no funcionales

Anexo K. Árbol escena 3D

Anexo L. Storyboard

Anexo M. Modelo casos de uso

Anexo N. Diseño e Implementación

Anexo O. Manual de instalación y mantenimiento

Anexo P. Manual de usuario

GLOSARIO

3D: en geometría y análisis matemático, un objeto o ente es tridimensional si tiene tres dimensiones. Es decir cada uno de sus puntos puede ser localizado especificando tres posiciones en un sistema de coordenadas cartesianas (X, Y, Z).

AVATAR: es una representación tridimensional del usuario generalmente se representa con un modelo tridimensional con forma humanoide dentro del entorno virtual, este efectúa las diferentes acciones que le indica el usuario mediante un periférico como un ratón (*Mouse*) o el teclado. Las acciones como caminar, saltar, correr, entre otras, pueden ser representadas por un avatar.

EV: por sus iniciales, Entorno Virtual. Entorno sintético que representa por medio de imágenes tridimensionales y en tiempo real un ambiente sea real o imaginario.

INMERSIÓN: acto voluntario de obviar todos los estímulos que indican que la experiencia que se presenta no es real y, por tanto, acaparar toda la concentración y atención de la persona involucrada.

INTERACCIÓN: tener control del sistema creado. Para lograr la interacción existen diversas técnicas e interfaces hombre-máquina, que van desde teclado y Mouse hasta guantes o trajes sensoriales. La interactividad con el mundo virtual supone que el usuario pueda mover objetos (además de a sí mismo) y modificarlos, y que tales acciones produzcan cambios en ese mundo artificial.

JMONKEY ENGINE (jME): es uno de los motores de juego más completo totalmente escrito y basado en tecnología JAVA. Usa una capa de abstracción que permite cualquier API de renderizado, adosable en forma de plug-in. Actualmente soporta LWJGL y JOGL. jME es código abierto, bajo licencia BSD. Está basado en una arquitectura de escenas tipo árbol. Esto permite la organización de los datos del juego en un grafo de nodos, donde un nodo padre puede contener cualquier número de nodos hijos, pero un nodo hijo solo contiene un único padre. Estos nodos están organizados espacialmente para permitir sencillamente descartar ramas de la estructura.

LIBRERÍA: conjunto de subprogramas que contienen código y datos que proporcionan servicios a programas independientes, utilizados para desarrollar software. Esto permite que el código y los datos se compartan y puedan modificarse de forma modular.

OBJETO 3D: es una representación en el entorno virtual de un objeto real o imaginario, el cual puede ser trasladado, rotado, escalado y/o modificado. Además, contiene atributos asociados como comportamiento, propiedades físicas y cinemáticas.

REALIDAD VIRTUAL: es una experiencia interactiva e inmersiva en un mundo simulado.

RECORRIDO VIRTUAL: permiten al usuario el desplazamiento tridimensional por el escenario sintético recreado o imaginado.

RENDER: proceso de computación 3D donde a partir de un modelo, se genere por computador una imagen ya sea imagen 3D o animación 3D. El modelo es una descripción en tres dimensiones de objetos en el lenguaje o estructura de datos estrictamente definidos. El modelo debe contener geometría, punto de vista, textura e información de iluminación. La imagen resultado de la renderización es una imagen digital.

RIESGO: probabilidad de que, en una situación dada, un agente externo produzca un daño. Para que exista un riesgo, es necesario que se esté expuesto a un peligro. Se necesitan tanto el peligro como la exposición, si alguno de ellos no se presenta, entonces no hay riesgo.

SOFTWARE LIBRE: (en inglés free software) es la denominación del software que brinda libertad a los usuarios sobre su producto adquirido y por tanto, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente.

RESUMEN

El presente trabajo de investigación y desarrollo, emplea metodologías del mundo de la ingeniería de software, ajustadas adecuadamente para el desarrollo de entornos virtuales, así como también herramientas típicas para soportar procesos de desarrollo estudiadas en el transcurso de formación de los estudiantes de Ingeniería Informática de la Universidad Autónoma de Occidente (UAO).

Este proyecto está basado en una serie de trabajos previos relacionados con el recorrido virtual por la UAO y Virtual Campus 3D a cargo del grupo de investigación en Telemática e Informática Aplicada de la Universidad Autónoma de Occidente.

Las actividades para recrear un evento riesgo y su plan de evacuación normalmente se realizan al aire libre, pero estas se ven limitadas por presupuestos, tiempo, espacios y medio ambiente haciendo que la información obtenida a partir de cada situación de riesgo presentada sean apreciaciones propias del momento, sin generalizarlas. Haciendo un estudio de estas necesidades se comprobó que en nuestro país no se han popularizado aún los prototipos de software que representan estas actividades.

Virtual Campus 3D- Risks se desarrolla como plataforma para responder a la necesidad planteada, entregando al usuario el contenido necesario para poder comprender en forma simple, práctica, entretenida y general, el plan de evacuación de las instalaciones de la UAO. Está cimentado en software libre y contiene dos módulos, un módulo cliente que permite crear un recorrido virtual por la UAO con un guía inteligente quien se encarga de nombrar características del diseño arquitectónico del campus como el arco de entrada, edificio central, plaza académica, ágora y sótano; indica las zonas de refugio y rutas de evacuación, que son claves para identificar la contramedida a los riesgos que pueden ocurrir en las instalaciones de la Universidad. Y un módulo administrativo que se integra con el portal de la Universidad, encargado de gestionar los diferentes componentes del recorrido como las imágenes, barra de menú, animaciones y objetos 3D.

La plataforma sigue buenas prácticas de la ingeniería del software y para su desarrollo se hizo uso de una metodología particular para este contexto de aplicación denominada UP4VED; esto permite, contar con una plataforma escalable que puede evolucionar con el tiempo y ajustarse a nuevas necesidades tanto para el módulo cliente como para el administrativo.

INTRODUCCIÓN

El campo primordial a abordar en cualquier tipo de riesgo al cual se exponga un ser humano en cualquier escenario, debe ser el de la Seguridad, factor que aún no presta suficiente atención. Dos de las ramas de la Seguridad son: La seguridad física y salud ocupacional, las cuales no son tomadas en cuenta sino hasta que sucede un primer riesgo que se trata de corregir con el *después de*.

A partir de cada una de estas ramas se ha encontrado que para cualquier caso de emergencia se deben tener en cuenta en general la ubicación actual del afectado, las rutas de evacuación y zonas de refugio. Que en el caso de la Universidad están actualmente designadas.

Para el caso de la rama de la seguridad física, la Universidad cuenta con la cultura de sus integrantes como el uso del Carnet, el cuidado de objetos personales, uso de la calcomanía en vehículos y el uso exigente de Carnets a visitantes.

Y, ¿Qué es un Riesgo? Un riesgo se puede definir como la probabilidad de que, en una situación dada, un agente externo produzca daño. Para que exista un riesgo, es necesario que se esté expuesto a un peligro. Se necesitan tanto el peligro como la exposición; si alguno de ellos no se presenta, entonces no hay riesgo.

Entonces a partir de esto, se implementó una plataforma para el desarrollo de entornos virtuales, que permiten la visualización de eventos que representan riesgos y su respectivo plan de evacuación por medio de un tour guiado a través de la arquitectura de la UAO y que tiene como destinatarios a los miembros de la institución educativa.

La simulación de eventos dentro del campo de los Entornos Virtuales posibilita el aprendizaje ofreciendo al usuario un escenario de práctica muy similar al de su realidad incentivando su formación. Desde la observación, se percibe que, durante las prácticas físicas realizadas para la prevención de un riesgo, se han presentado inconvenientes y accidentes, poniendo en peligro a los implicados, y todo esto se hace con el fin de llevar a cabo un objetivo de evaluación de contramedida frente a un riesgo presentado.

Por lo tanto, la plataforma desarrollada define el cómo realizar este último objetivo sin escatimar las posibilidades de entendimiento de los capacitados y de forma que éstos no se vean afectados durante la ejecución del plan contingente, sino que sirve como base de entrenamiento para la prevención y comprensión global de lo que el Departamento de Protección y Control planteó.

El desarrollo de una plataforma para la creación de entornos virtuales, al igual que cualquier tipo de software, requiere de una metodología, estrategias y ciclos de desarrollo. En este desarrollo se plantearon nuevas funciones al módulo Web (Administrativo) y una expansión de objetos que interpretan acciones de riesgos e inteligencia artificial por controladores al módulo cliente en Virtual Campus 3D creado por el Grupo de Investigación en Telemática e Informática Aplicada (GITI). Esto fue posible gracias a la arquitectura implementada y orden del código fuente.

1. PLANTEAMIENTO DEL PROBLEMA

Actualmente, los seres humanos se encuentran expuestos a cientos de contextos tecnológicos que nos han obligado a cambiar los criterios de legitimidad de los conocimientos y por supuesto los de formación en las personas. A las organizaciones se les exige la formación de sus profesionales al mismo tiempo que los medios para facilitar y optimizar ese proceso de formación donde la simulación juega un papel muy importante.

En los países en vías de desarrollo es común la carencia de herramientas para completar procesos educativos, que se presentan en el área de informática y de desarrollo de software en común, esto excluye a que otras dependencias de una misma organización, no se familiaricen con las tecnologías de la información.

Las visualizaciones reales que, a menudo los Departamentos de Salud Ocupacional y de Protección y Control realizan, resultan ser una pequeña versión o percepción de lo que a largo plazo sería la contramedida realizada a partir de un evento de emergencia que se pueda presentar en la UAO y que, además generan costos de implementación y contratación de organismos de seguridad, por cada evento que se quiera recrear durante una capacitación.

Es evidente que, las instituciones educativas, en general, no tienen un prototipo software de contramedidas a posibles eventos de riesgo, lo que implica que la capacitación realizada a un grupo nuevo de la brigada de seguridad de la UAO, se vea limitada a que un miembro tome apuntes o a tenga una percepción propia y no general de la situación presentada.

A partir de esto, se investigó la posibilidad de implementar un entorno virtual, el cual debía brindar una alternativa para solucionar la necesidad, ofreciendo un espacio de aprendizaje, dando una mejor perspectiva, preparando al miembro de la brigada y a quienes se vean involucrados en el evento-riesgo presentado, sin sufrir ningún tipo de accidente, mejorando la actuación frente a una emergencia en donde se vea obligado a seguir los procedimientos de evacuación establecidos y reduciendo al 100% la generación de desperdicios y gases creados por algún tipo de explosión y/o incendio que en la realidad, han causado efectos y/o impactos al medio ambiente.

Se utilizó como base la plataforma Virtual Campus 3D (VC3D) creada por investigadores del grupo de investigación en telemática e informática aplicada. La implementación del EV, se fundamentó en el motor de desarrollo jMonkey Engine empleando el lenguaje de programación JAVA, utilizado por los Ingenieros Autónomos y a nivel mundial como herramienta libre de fácil aprendizaje.

Gracias a los entornos virtuales, se puede recrear y simular situaciones ligadas a la realidad, con el agregado de que el usuario del entorno virtual interactúe y

manipule la escena sin preocuparse por los riesgos que representa estar en la situación, al tiempo que se adquieren las capacidades y competencias para los niveles de formación de aquellos que realizan estas tareas en casos de emergencia.

2. JUSTIFICACIÓN

El uso de una plataforma adecuada para el desarrollo de entornos virtuales que permita la visualización 3D de procesos de evacuación y prevención a emergencias, fue considerado como un punto estratégico para el acercamiento a la descripción de la funcionalidad del sistema.

Las herramientas utilizadas para la visualización 3D, son útiles en escenarios de educación técnica, tecnológica e ingeniería. La flexibilidad para crear diferentes ambientes en un sistema dado y la intervención que se puede realizar sobre su código, hace de estas plataformas unas excelentes herramientas para desarrollar nuevos proyectos por parte de los interesados en este tipo de investigaciones.

Dentro de las ventajas de elaborar entornos virtuales para la visualización de riesgos y los procedimientos a realizar dentro de estos, se tienen:

- Realizar un proceso de aprendizaje, para las diferentes partes que intervienen en estos eventos: como personas, profesionales del oficio y otros. (miembros de la universidad y otros que quieran conocer el campus)
- La visualización permite dar a conocer los riesgos que pueden ocurrir dentro de las instalaciones de la Universidad y la adecuada evacuación de los involucrados.
- La visualización facilita el aprendizaje de quienes se pueden ver envueltos en una de las situaciones presentadas.
- El uso de entornos virtuales puede ser presentado a una multitud de personas, disminuyendo el costo y riesgo que representa el hacer un experimento real dentro del Campus.

Es posible continuar con el proceso de robustecer la plataforma gracias a su arquitectura, que es extensible y capaz de adaptarse a nuevas adiciones con muchas más funcionalidades que cubran otros aspectos de la seguridad, que no se cubrieron en esta fase.

3. ANTECEDENTES

En Estados Unidos se ha tratado el tema del uso de entornos virtuales como espacio de inducción a trabajadores que reciban estos métodos innovadores de aprendizaje que sirven como contramedida en los casos que representen riesgos.

El Servicio de Extensión de Ingeniería de Texas (TEEX), es un proveedor de formación para la agencia federal, local y personal de emergencia, así como muchos otros tipos de programas, se enfoca en la búsqueda de oportunidades de promoción y formación en entornos virtuales. TEEX revisa los proyectos recientes, incluyendo una simulación de un diseño de un puente que enseña a los trabajadores de mantenimiento a identificar los factores que causan roturas de puentes. También incluirán un nuevo proyecto de formación para la búsqueda y rescate urbano, haciendo buen uso de los mundos virtuales (Virtual World, VW) donde se puede utilizar para la colaboración, conferencias, asistencia técnica y capacitación para casos de emergencia de países de todo el mundo.

Code3D aparece como alternativa también, pues es muy dinámica su programación, como si se estuviera dirigiendo el guión de una película; aunque se dispone de mucho tiempo para generarlo, con otras características anexas como los modelos 3D ya creados. Es un producto de software de simulación de código abierto en 3D que ha sido diseñado para ayudar a los instructores de los servicios de emergencia para que sirva como contramedida a una gran variedad de situaciones (materiales peligrosos, incendios, policía, etc.) en un ambiente seguro antes de que se efectúe la experiencia en el mundo real. Los usuarios pueden crear escenarios para satisfacer sus propias necesidades o utilizar y personalizar uno de los escenarios pre-construidos. Code3D es una herramienta más que los servicios de emergencia pueden agregar a sus entidades de capacitación¹

En Colombia, existen entidades como La Prevención y Atención de Desastres, la Defensa Civil y otros, que simulan eventos coordinados sobre los posibles riesgos y emergencias que puedan ocurrir dentro de alguna entidad, realizándolas de forma presencial, real. Esto nos indica que, aún, no se ha presentado una propuesta de control de emergencias en 3D para realizar los pasos requeridos, en cuanto a crear una contramedida por el riesgo de tipo de seguridad física presentado; es ésta entonces, una investigación que satisface el inicio de una nueva perspectiva en el campo del desarrollo de entornos virtuales que están en la capacidad de ilustrar un personal de seguridad de cualquier entidad.

¹ SKOPIC, Emily. Welcome to Code 3D Interactive Training [en línea]. Code 3D. [Consultado 15 de Febrero de 2011]. Disponible en internet: <URL:<http://code3d.com/>>

Desde el punto de vista de la observación, se percibe que, durante las prácticas físicas realizadas para la prevención de un riesgo, se han presentado inconvenientes y accidentes, poniendo en peligro a los implicados, y todo esto se hizo con el fin de llevar a cabo un objetivo de evaluación de contramedida frente a un riesgo presentado. Por lo tanto, en este proyecto, se precisó el cómo realizar este último objetivo sin escatimar las posibilidades de entendimiento de los capacitados y donde éstos no se ven afectados durante la ejecución del plan contingente, sino que sirve como base de entrenamiento para la prevención y comprensión global de lo que el Departamento de Protección y Control requiere; además de la implementación de un tour guiado y recorrido virtual que se ha venido planteando con otros trabajos anteriores del grupo GITI.

4. MARCO TEÓRICO

Teniendo en cuenta que nuestra realidad se encuentra un poco lesionada gracias a sus múltiples intentos de modernización, la tecnología ha tratado de cubrir las promesas contenidas en esta era digital y ha suplido esa necesidad de crecimiento, haciendo que las operaciones del mundo real se simplifiquen en alto porcentaje con el uso masivo de las nuevas tecnologías de la información y la comunicación. Dentro de ese ámbito de tecnologías y sistemas de información se puede encontrar el término realidad virtual, tecnología que ha tenido mucho eco desde hace más de cuatro décadas atrás surgiendo de una sola idea que haría posible la interacción de personas con computadores de una forma totalmente revolucionaria.

La definición de realidad virtual es una tarea un poco más compleja, existen tantas definiciones como investigadores que trabajan con ella y dentro de las mismas se denomina a la RV de diversas maneras: entornos sintéticos, ciberespacio², realidad artificial o mundos artificiales. Otros autores señalan el carácter paradójico (*oxymoron*) de la expresión Realidad Virtual³⁻⁴, que denota “realidad no real”⁵. Gracias a que es una herramienta complementaria en cualquier área, su uso en determinadas aplicaciones facilita la comprensión de la información obtenida y para otras aplicaciones despierta el interés por el aprendizaje; donde fácilmente se pueden recrear la escena de una emergencia provocada en un lugar determinado, desde miles de lugares distintos sin impacto ambiental ni territorial.

La plataforma desarrollada hace uso de algunas características de la realidad virtual, contiene actividades relacionadas con gráficos, modelos y visualización de objetos tridimensionales y animaciones, incluyendo sistemas de interacción 3D en el uso de plataformas y motores de juegos que permitieron la recreación de algunas actividades y operaciones de la vida real, como la exposición al fuego, la visualización de las rutas de evacuación y zonas de refugio, mencionados en el plan de evacuación de la UAO.

² El término ciberespacio acuñado por William Gibson en su novela *Neuromancer* en el año 1984, es usado para englobar el conjunto de aplicaciones que comparten como plataforma las redes de comunicación.

³ KRUEGER, Myron W. Responsive Environments. In NCC Proceedings, 1977; p. 375-385.

⁴ ISDALE, Jerry. What is Virtual Reality? [en línea]. Reino Unido: Versión 4 – Draft 1, Septiembre, 1998. [Consultado 12 de Febrero de 2011]. Disponible en Internet: <URL:<http://vr.isdale.com/WhatIsVR/frames/WhatIsVR4.1.html>>

⁵ CARDONA, Jesús David; HIDALGO, Miguel Ángel; CASTÁN, Héctor; ROJAS, Fabio; BORRO, Diego; JARAMILLO, Héctor. Realidad Virtual y Procesos de Manufactura: Prototipado, simulación y realidad virtual. Cali: Universidad Autónoma de Occidente, 2007; p. 15.

4.1 GRÁFICOS 3D

Normalmente cuando las personas interactúan con el mundo real la percepción obtenida es totalmente tridimensional, mientras que las imágenes visuales que se ven a través de los ojos humanos a diferentes pantallas, son bidimensionales.

P. ej.: en un monitor de computador se perciben dos dimensiones reales: alto y ancho. Pero cuando observan películas como *“Toy Story”* y videojuegos como *“Tomb Raider”* utilizando la misma interfaz, se percibe un mundo de tres dimensiones⁶.

Una particularidad sobresaliente de estos gráficos 3D es que el mundo ilustrado, puede ser una representación del mundo en que se vive, en el que se vivirá en el futuro o un mundo que vive únicamente en las mentes de los creadores de películas o videojuegos. Actualmente los gráficos 3D han planteado un problema que va más allá de la inventiva natural del hombre, este problema es que se espera un alto grado de realismo en cada cosa que el ser humano ve.

A mediados de 1970 un juego llamado *“Pong”* impresionó a las personas por sus gráficos en pantalla; hoy, el mercado de videojuegos está a la vanguardia con películas en formato DVD y se espera que los juegos sean suaves y detallados tanto como las imágenes que se aprecian en las salas de cine.⁷

4.1.1 ¿Qué son gráficos 3D? Una imagen que tiene o aparenta tener alto, ancho y profundidad se puede decir que es tridimensional o 3D. Los gráficos 3D son todos los objetos que se pueden dibujar en un espacio de tres ejes como puntos (dimensión 0), curvas (dimensión 1), superficies (dimensión 2) y otros objetos sólidos (dimensión 3) formados por caras poligonales⁸.

Para muchas personas los sistemas avanzados de juegos son las formas más comunes de ver gráficos 3D. Esos sistemas hechos con imágenes generadas por computador, pasan a través de tres grandes pasos para crear y representar una escena 3D realística: crear un mundo 3D virtual, determinar que parte del mundo

⁶ FRANKLIN, Curt. ¿How 3-D Graphics Work? [en línea]. Estados Unidos, 2000. How Stuff Works? Discovery Company. [Consultado 27 de Enero de 2011]. Disponible en Internet: <URL:<http://computer.howstuffworks.com/3dgraphics.htm>>

⁷ Ibíd., Disponible en Internet: <URL:<http://computer.howstuffworks.com/3dgraphics.htm>>

⁸ ABREU LEÓN, Jose Luis. Gráficos 3D [en línea]. España: Ministerio de Educación, 2011. [Consultado 27 de Enero de 2011]. Disponible en Internet: <URL:http://newton.cnice.mec.es/Documentacion_3/Doc/Graficos_3D.html>

será mostrado en la pantalla y determinar cómo se ve cada pixel en la pantalla para que toda la imagen sea lo más real posible⁹.

4.1.2 Características. Los gráficos 3D se caracterizan porque además de representar geometrías e imágenes 2D en perspectiva, puede hacer que el espectador interactúe con estos gráficos (así no sean objetos del mundo real), utilizan técnicas como iluminación y animación para alcanzar otros efectos de profundidad. Un sistema de gráficos 3D necesita enfrentar muchos problemas cuando se representan los objetos gráficos y sus propiedades, facilitando transformaciones, organizando todos los componentes y renderizando la escena.

Estas son las características más sobresalientes de los gráficos 3D:

- **Modelado.** A partir de esta función se crean y modifican figuras diseñadas por el artista además de las primitivas (p.ej. esferas, conos, cajas, etc.) haciendo que la transformación de estos objetos ocupe el lugar más importante en un sistema de gráficos. Las transformaciones geométricas son usadas para ubicar los objetos geométricos en el mundo virtual y para cambiar sus formas, tamaños y ubicaciones cuando es necesario.
- **Iluminación.** Un objeto gráfico va mucho más allá de ser una representación geométrica pues contiene las propiedades de apariencia que determinan cómo el objeto será renderizado. Esas propiedades incluyen colores, texturas y propiedades de materiales para un sombreado más sofisticado. Las luces o iluminación y los sombreados controlan la forma en que son calculados los colores y las intensidades de luz que recaen sobre un objeto.
- **Renderizado.** El renderizado 3D no está limitado a una sola escena estática. Los efectos dinámicos causados por procesos de renderizado pueden incluir animación e interacción. Interacción es una alteración de la escena y es resultado de operaciones realizadas por el usuario al escenario o directamente con objetos del mismo; la animación es un cambio diseñado internamente en el mundo virtual.

4.1.3 Creación de gráficos 3D. Los objetos 3D en un mundo virtual, no existen físicamente, son totalmente sintéticos. Las únicas propiedades que tienen estos objetos 3D son agregados por software. Los programadores usan herramientas

⁹ FRANKLIN, Curt. ¿How 3-D Graphics Work? [en línea]. Estados Unidos, 2000. How Stuff Works? Discovery Company. [Consultado 27 de Enero de 2011]. Disponible en Internet: <URL:<http://computer.howstuffworks.com/3dgraphics.htm>>

especiales y definen un mundo virtual con muchos detalles para que todo en él siempre se comporte de la forma en que sea un modelo adaptable al espectador.

Los gráficos 3D estudian el modelado y renderizado de un espacio 3D. Los objetos tienen diferentes clases de propiedades materiales, donde algunas fuentes de luces de varias características iluminan la escena del espacio virtual. Las cámaras virtuales o también llamadas visores 3D capturan las escenas del mundo virtual y pueden ubicarse en diferentes lugares en el espacio aún cuando poseen diferentes características.¹⁰

En un momento dado, la pantalla muestra únicamente una pequeña parte del mundo virtual creado por computador. Lo que se muestra en la pantalla se determina por la forma en que está definido el mundo, donde se elige el camino o ruta y la vista. No importa a donde va, el mundo virtual determina lo que el espectador verá desde esa posición y la vista en esa dirección. Cada objeto debe verse y moverse de forma que convenza al espectador que este objeto siempre tiene la misma consistencia: que es duro o suave, rígido o plegable, etc.¹¹

4.1.4 ¿Cómo se clasifican? La mayoría de personas conocen el mecanismo por el que se generan los gráficos 2D, y aunque para algunos, el resultado final con gráficos 2D y 3D suele ser una imagen bidimensional en pantalla o en impresora, el proceso de control de los gráficos 3D es muy distinto y mucho más complejo. Una idea que debe quedar clara es la diferencia que existe entre los gráficos 2D y 3D generados por computador y en tiempo real, que básicamente consiste en que unos "viven" en un mundo plano y los otros "habitan" en un mundo espacial, lo que no es un gran problema para un computador, ya que en principio, exceptuando el aumento de la cantidad de datos a procesar (y los consecuentes aumentos del tiempo y el espacio de proceso), da casi igual manejar puntos con dos coordenadas que con tres, salvo por lo que respecta a la mayor o menor integración de los elementos que los manejan.¹²

- **Gráficos 3D por computador.** Son gráficos que utilizan una representación tridimensional a partir de un dato geométrico que se guarda en el computador con el propósito de realizar cálculos y representación de imágenes 2D.

¹⁰ ZHANG, Hong; LIANG, Daniel. Computer graphics using Java 2D and 3D: Basic 3D Graphics [en línea]. Estados Unidos: Prentice Hall, 2007. [Consultado 28 de Enero de 2011]. Disponible en internet: <URL: <http://books.google.com.co/books?id=IJjKSRS558C>>

¹¹ FRANKLIN, Curt. ¿How 3-D Graphics Work? [en línea]. Estados Unidos, 2000. How Stuff Works? Discovery Company. [Consultado 27 de Enero de 2011]. Disponible en Internet: <URL: <http://computer.howstuffworks.com/3dgraphics.htm>>

¹² MONTERO, Ramón. Conceptos básicos de gráficos 3D [en línea]. España, 1995. ©2011 IDG COMMUNICATIONS, S. A. U. [Consultado 28 de Enero de 2011]. Disponible en internet: <URL: <http://www.idg.es/pcworld/Conceptos-basicos-de-graficos-3D/art33575.htm>>

Existe una gran dificultad cuando se trata de trabajar con gráficos 3D generados con el computador, ya que se debe proyectar unas entidades de tres dimensiones en la memoria unidimensional que posee el computador, además realizar cálculos geométricos complejos con una herramienta secuencial y conseguir resultados válidos para el mundo real.

El computador en lugar de almacenar la información sobre puntos, líneas y curvas en planos bidimensionales, almacena la posición de puntos, líneas y caras (para construir un polígono) en un espacio de tres dimensiones. El software actual para generación de gráficos va más lejos de sólo el almacenamiento de polígonos en la memoria del computador. Las gráficas de hoy además de ser colecciones masivas de polígonos en formas reconocibles, también pueden resultar de técnicas que emplean *Shading* (Sombreadores), *texturing* (Texturizado o mapeado) y la *rasterización* (En referencia a mapas de bits)¹³

- **Gráficos 3D en tiempo real.** Los gráficos 3D en tiempo real se generan bajo demanda, es decir, cuando un usuario solicita la acción “girar a la izquierda” en el mundo 3D, el procesador en tiempo real o *motor* crea los gráficos necesarios para desplazar la vista hacia la izquierda. Las imágenes pre-representadas no disfrutan de ese lujo, ya que el artista las genera mucho antes de que el espectador pueda verlas. El inconveniente radica en que las imágenes pre-representadas poseen una profundidad de color, complejidad y efectos especiales mucho más ricos, mientras que los gráficos en tiempo real suelen tener un nivel de detalle muy reducido, colores y texturas muy limitados y efectos especiales mínimos.

Cada vez hay más consumidores que ven este tipo de gráficos 3D en tiempo real en sus vidas cotidianas, ya sea por la televisión o en los últimos juegos 3D que salen al mercado. Los sistemas de creación de juegos para consolas que admiten la función de tiempo real todavía están en su primera generación. Gracias a la repentina popularidad de las tarjetas de videojuegos 3D en tiempo real para PC, los desarrolladores están produciendo cada vez más juegos en los que se utilizan gráficos 3D en tiempo real¹⁴.

El modelado y la elaboración de texturas realistas son los factores clave y la parte más importante de la creación de gráficos para juegos en tiempo real. La infinidad de aspectos y detalles técnicos que implica la creación de un modelo eficiente (con un número reducido de polígonos), que sea creíble, estéticamente atractivo y

¹³ *Ibíd.*, Disponible en internet: <URL:<http://www.idg.es/pcworld/Conceptos-basicos-de-graficos-3D/art33575.htm>>

¹⁴ S.A.B.I.A. Sistemas Adaptativos y Bioinspirados en Inteligencia Artificial. Modelado para juegos 3D en tiempo real [en línea]. Coruña, España. [Consultado 02 de Febrero de 2011]. Disponible en internet:<URL:<http://sabia.tic.udc.es/gc/Contenidos%20adicionales/trabajos/Juegos/modelado%203d%20juegos/index.htm>>

esté preparado para comportarse y presentarse correctamente en el mundo en tiempo real son problemas a los que tiene que enfrentarse únicamente el artista. Después de todo, los motores en tiempo real sólo se basan en los elementos más básicos de los gráficos 3D para crear la ilusión. El modelado de gráficos en tiempo real es un proceso muy delicado. Hay que saber cuál va a ser exactamente el resultado final, que se va a obtener después de exportar el objeto modelado al motor de tiempo real. Cuantas más cosas se conozcan acerca de cómo "piensa" un motor en tiempo real típico, mejores serán los resultados del trabajo inicial, ahorrando tiempo y frustraciones¹⁵.

Los gráficos 3D en tiempo real y los gráficos 3D pre-representados de alta gama tienen muchos elementos en común. Sin embargo, para conseguir la velocidad necesaria para un juego, con los gráficos en tiempo real se deben usar únicamente los elementos indispensables; es decir, la geometría, las transformaciones y las propiedades de superficie de la malla. La mayoría de las veces, el que crea estos elementos es el programa de exportación (es decir, la aplicación de una tercera entidad que traduce el modelo original a un lenguaje que pueda comprender el motor de juegos). Luego, estos elementos se colocan en algún tipo de archivo (o en lenguaje de programación, antes de compilarlo para generar código binario), para que puedan ser editados manualmente, si es necesario. En ocasiones, estos elementos se pueden dividir en varios archivos independientes, uno para la geometría, otro para las propiedades de superficie y otro para las transformaciones; las cuales se combinan cuando se compila el archivo que recibe el motor de juegos¹⁶.

La mayoría de las interfaces de programación de aplicaciones (API, Application Programming Interface) para gráficos 3D en tiempo real proporcionan un convertidor propio que funciona con formatos conocidos, como 3DS o DXF. En muchos sitios web de Internet hay disponibles convertidores de distribución gratuita para exportar archivos OBJ o VRML. La API de un motor en tiempo real suele constar de un conjunto de herramientas, comandos, o ambas cosas, para que los desarrolladores independientes puedan trabajar con el producto de una determinada empresa. En los juegos en tiempo real, la API permite a los desarrolladores de juegos emplear el motor de juegos 3D como base sobre la que crean sus programas usando sus propios comandos personalizados, que se comunican con el hardware a través de la API¹⁷.

¹⁵ Ibíd., Disponible en internet: <URL:<http://sabia.tic.udc.es/gc/Contenidos%20adicionales/trabajos/Juegos/modelado%203d%20juegos/index.htm>>

¹⁶ MONTERO, Ramón. Conceptos básicos de gráficos 3D [en línea]. España, 1995. ©2011 IDG COMMUNICATIONS, S. A. U. [Consultado 28 de Enero de 2011]. Disponible en internet: <URL:<http://www.idg.es/pcworld/Conceptos-basicos-de-graficos-3D/art33575.htm>>

¹⁷ MONTERO, Ramón. Conceptos básicos de gráficos 3D [en línea]. España, 1995. ©2011 IDG COMMUNICATIONS, S. A. U. [Consultado 28 de Enero de 2011]. Disponible en internet: <URL:<http://www.idg.es/pcworld/Conceptos-basicos-de-graficos-3D/art33575.htm>>

4.2 INTERACCIÓN 3D

La interacción es una acción que se ejerce recíprocamente entre dos o más objetos, agentes, fuerzas, funciones, etc. La interacción entre el computador y el espectador es un proceso cíclico en tres estados: escuchar, pensar y responder; donde es imprescindible realizar las tres operaciones y realizarlas bien¹⁸.

El espacio 3D para la interacción puede ser el espacio físico real, un espacio virtual simulado en el computador, o una combinación de ambos. Cuando el espacio real se utiliza para la entrada de datos, los seres humanos realizan acciones o dan órdenes a la máquina utilizando un dispositivo de entrada que detecta la posición en 3D de la acción humana. Cuando se utiliza para la salida de datos, la simulación de la escena virtual en 3D se proyecta en el entorno real a través de un dispositivo de salida¹⁹.

En la interacción en 3D, los usuarios llevan a cabo sus tareas y funciones de intercambio de información con los sistemas informáticos en el espacio 3D. Es un tipo de interacción intuitiva porque los seres humanos interactúan en tres dimensiones con el mundo real. Los usuarios experimentan una sensación de presencia cuando participan en un mundo virtual inmersivo; permitiéndoles interactuar con ese mundo en 3D, el cual les permite hacer uso del conocimiento natural e intrínseco de cómo el intercambio de información se lleva a cabo con los objetos físicos en el mundo real. La textura, el sonido y el habla pueden ser utilizados para aumentar la interacción 3D²⁰.

Están también los sistemas de interacción 3D que son sistemas informáticos que permiten al usuario experimentar libremente con contenidos en un espacio tridimensional y que responden a sus acciones de manera instantánea. Algunos ejemplos de sistemas interactivos son los videojuegos, museos virtuales, demostradores, visualización de arquitectura, comunidades virtuales 3D, simuladores, campos virtuales, experimentación artística, cine de animación, etc.²¹

La Figura 1 ilustra el flujo de trabajo de una interacción 3D.

¹⁸ CRAWFORD, Chris. On game design [en línea]. New Riders. 2003. [Consultado 02 de Febrero de 2011]. Disponible en Internet: <URL:<http://books.google.com.co/books?id=0BQFquvAC8C>>

¹⁹ PARKER, Philip M. Extended Definition: 3D interaction [en línea]. Francia. [Consultado 02 de Febrero de 2011]. Disponible en Internet: <URL:<http://www.websters-dictionary-online.com/definitions/3D+Interaction?cx=partner-pub-0939450753529744%3Av0qd01-tdlq&cof=FORID%3A9&ie=UTF-8&q=3D+Interaction&sa=Search#922>>

²⁰ Ibíd.,. Extended Definition: 3D interaction.

²¹ TAIBO, Javier. Entornos tridimensionales interactivos [en línea]. Coruña: España, Universidad de la Coruña, 2009. [Consultado 02 de Febrero de 2011]. Disponible en Internet: <URL:http://videalab.udc.es/i3d/docs/clases/2009_2010/I3D_Introduccion.pdf>

Figura 1. Flujo de trabajo de una interacción 3D



Fuente: TAIBO, Javier. Entornos tridimensionales interactivos [en línea]. Diapositiva 29. [Consultado 02 de Febrero de 2011]. Disponible en internet: <URL:http://videalab.udc.es/i3d/docs/clases/2009_2010/I3D_Introduccion.pdf>

4.3 REALIDAD VIRTUAL Y ENTORNO VIRTUAL

Los entornos virtuales surgieron a partir de la era de la simulación en la década de 1960. Los entornos virtuales son pantallas interactivas que por medio de un computador dan la ilusión de desplazamiento a otro lugar. A esta ilusión se le han dado términos como el *oxímoron* “realidad artificial” y “realidad virtual”, que sugieren un rendimiento mucho más alto que la tecnología actual puede proporcionar. Expresiones como “mundos virtuales” y “entorno virtual” parecen preferibles porque son lingüísticamente más conservadores a diferencia del desconcertante neologismo “ciberspacio”²². Los entornos virtuales se pueden definir como pantallas interactivas, como imágenes virtuales mejoradas usando modalidades de visualización auditiva y táctil, para convencer a los usuarios que se encuentran inmersos en un espacio sintético²³.

²² R. ELLIS, Stephen. ¿What are virtual environments? [en línea]. Washington, IEEE Computer Graphics and Applications, 1994. vol. 14, no. 1, p. 17-22. [Consultado 07 de Febrero de 2011]. Disponible en internet: <URL:<http://www.computer.org/portal/web/csd/doi/10.1109/38.250914>>

²³ Ibíd., ¿What are virtual environments?: Abstract

4.3.1 Definición de realidad virtual. La realidad virtual no es más que un concepto, una idea, algo similar a la “dimensión fractal” o la “vida artificial”. Hay quienes definen la realidad virtual como “una sofisticada interface de comunicación hombre-máquina”. Otros la califican como “un simulador avanzado interactivo”. También la han definido como: “la tecnología que nos permite formar parte de un mundo artificial”²⁴.

La dificultad de encontrar una definición concreta proviene de la variedad de aplicaciones que se derivan o han derivado de ella, y que en su propio nombre está la principal raíz de todas las dudas, pues “realidad virtual”, si se hace una aproximación a sus términos, significaría algo como: “realidad no real”²⁵.

La realidad virtual (RV) es una combinación de conceptos donde se puede decir que es una simulación tridimensional generada o asistida comúnmente por computador de algún aspecto del mundo real o ficticio, en el cual el espectador tiene la sensación de pertenecer a ese ambiente sintético o interactuar con él. La RV permite interactuar con mundos tridimensionales de una manera más natural, por ejemplo, un usuario puede realizar acciones dentro de un modelo virtual, desplazarse, moverse, caminar a través de él o levantar cosas, y de esta forma experimenta situaciones que se asemejan a las del mundo real.

“La utilización de la modelación y simulación que habilitan a la persona a interactuar con una visión tridimensional o a través de sensores ambientales. De tal forma que las aplicaciones emergen al individuo en ambientes generados por computador que simulan la realidad a través del uso interactivo de estos dispositivos, los cuales envían y reciben información”²⁶.

Las características de un sistema de realidad virtual, que lo distinguen de otros sistemas informáticos son:

- **La inmersión.** Propiedad mediante la cual el usuario tiene la sensación de encontrarse dentro de un mundo tridimensional.
- **Existencia de un punto de observación o referencia.** Que permite determinar la ubicación y posición de observación del usuario dentro del mundo artificial o virtual.
- **Navegación.** Propiedad que permite al usuario cambiar su posición de observación.

²⁴ MURILLO ALFARO, Félix. Realidad Virtual [en línea]. Lima, Perú: Instituto Nacional de Estadística e Informática (INEI). [Consultado 08 de Febrero de 2011]. Disponible en Internet: <URL:<http://www.inei.gob.pe/biblioineipub/bancopub/inf/lib5047/INDEX.htm>>

²⁵ Ibíd., Realidad Virtual. Definición.

²⁶ Encyclopædia Britannica. Virtual reality. Encyclopædia Britannica [en línea]. Encyclopædia Britannica, 2011. [Consultado 20 de Junio de 2011]. Disponible en Internet: <URL:<http://www.britannica.com/EBchecked/topic/630181/virtual-reality>>

- **Manipulación.** Característica que posibilita la interacción y transformación del medio ambiente virtual.

4.3.2 Definición de entorno virtual. Debido a que no todas las aplicaciones que se desarrollan pueden catalogarse como un sistema de realidad virtual, aparece por tanto, el concepto de Entorno Virtual (EV) que reúne aquellas aplicaciones que recrean en una pantalla de computador un espacio real o imaginario en 3D²⁷⁻²⁸, en donde no se busca la sensación de inmersión, sino simplemente la interacción²⁹ que puede obtener el espectador con el uso de medios tecnológicos.

El entorno virtual representa la metáfora de una generalización tridimensional en una bidimensional (o de escritorio). La principal innovación del concepto, establecido y elaborado en primer lugar por Ivan Sutherland y Myron Krüger era que la interfaz gráfica generada por computador pudiera convertirse en un ilusorio, sintético pero aparentemente un entorno físico. Estos entornos sintéticos pueden ser experimentados desde puntos de vista egocéntricos o exocéntricos. Es decir, los usuarios pueden parecer estar realmente inmersos en el entorno o verse a sí mismos representados a través de una ventana.³⁰

4.3.3 Realidad virtual y entornos virtuales: espacios virtuales. La simulación de un entorno interactivo no es más que el control de un modelo tridimensional generado por computador, de tal manera que permite la visualización e interacción en tiempo real. La representación del mundo virtual que se ve, se re-calcula constantemente, teniendo en cuenta la posición y el punto de vista del usuario, de manera que la imagen mostrada se ajusta a las imágenes obtenidas en el mundo real³¹.

A partir de los espacios virtuales, que pueden existir por separado, o también mezclados entre sí, se destacan los siguientes:

²⁷ STUART, Rory. The Design of Virtual Environment. New York: McGraw-Hill, 1996. 274 p.

²⁸ R. ELLIS, Stephen. ¿What are virtual environments? [en línea]. Washington, IEEE Computer Graphics and Applications, 1994. vol. 14, no. 1, p. 17-22. [Consultado 07 de Febrero de 2011]. Disponible en internet: <URL:<http://www.computer.org/portal/web/csdl/doi/10.1109/38.250914>>

²⁹ CARDONA, Jesús David; HIDALGO, Miguel Ángel; CASTÁN, Héctor; ROJAS, Fabio; BORRO, Diego; JARAMILLO, Héctor. Realidad Virtual y Procesos de Manufactura: Prototipado, simulación y realidad virtual. Cali: Universidad Autónoma de Occidente, 2007; p. 16.

³⁰ R. ELLIS, Stephen; R. BEGAULT, Durand; WENZEL, Elizabeth. Virtual Environments as Human-Computer Interfaces: Communication and Environments [en línea]. Moffet Field, California, E.U.A: NASA Ames Research Center, p. 164. [Consultado 08 Febrero de 2011]. Disponible en Internet: <URL:<http://books.google.com/books?id=WuQbERqXR10C>>

³¹ MURILLO ALFARO, Félix. Realidad Virtual [en línea]. Lima, Perú: Instituto Nacional de Estadística e Informática (INEI). [Consultado 08 de Febrero de 2011]. Disponible en Internet: <URL:<http://www.inei.gob.pe/biblioineipub/bancopub/inf/lib5047/INDEX.htm>>

- **El espacio virtual muerto.** Es aquel en el que no hay objetos en movimiento ni partes interactivas, permitiéndose sólo su exploración. Suele ser el que se observa en las animaciones tradicionales, donde las imágenes están pre calculadas y producen una experiencia pasiva.
- **El espacio virtual real.** Aquel en que los elementos tienen atributos reales, de tal manera que si se observa un reloj, este indique la hora; si se pulsan las teclas de una calculadora, se visualicen las operaciones aritméticas oportunas; si se abre un grifo, salga agua; y si la ubicación es una ciudad, se puedan ver personas y edificios.
- **El espacio virtual fantástico.** Permite realizar tareas irreales como, que un sujeto vuele o atraviese paredes. Es el típico entorno encontrado en los videojuegos, pero que también proporciona situaciones para las aplicaciones formales, como puede ser la posibilidad de observar un edificio donde el usuario vuele a su alrededor o de introducirnos dentro de un volcán.

4.3.4 Tipos de sistemas de realidad virtual (SRV). La calidad de la aplicación de la realidad virtual no depende de la sofisticación de los dispositivos que se empleen, sino de la capacidad de inmersión que se obtenga en el mundo virtual. Hay conceptos más importantes que otros para conseguir la sensación de estar dentro de un ambiente determinado. Por ejemplo, la velocidad de regeneración de las imágenes es más importante cuando se realiza un paseo con un casco virtual, que el que dicha imagen sea perfecta, ya que la visión no debe producir saltos.³²

Pero si se trata de analizar en una pantalla cómo se comporta el sistema de seguridad de un automóvil en caso de colisión, no importará que las imágenes se actualicen con cierta lentitud, a cambio de ver los detalles del proceso. Estas precisiones dan lugar a la aparición de múltiples tipos de sistemas virtuales, que se adaptan mejor a las características de los diferentes casos que se pueden considerar³³.

Si se precisan los elementos de entrada de datos, equipo necesario y visualizador de salida de un sistema virtual, estos sistemas se pueden clasificar en cuatro tipos principales:

- **Sistemas inmersivos.** Son aquellos sistemas donde el usuario se siente dentro del mundo virtual que está explorando. Este tipo de sistemas utiliza diferentes dispositivos denominados accesorios, como pueden ser guantes, trajes especiales, visores o cascos, estos últimos le permiten al usuario visualizar los

³² MURILLO ALFARO, Félix. Realidad Virtual [en línea]. Lima, Perú: Instituto Nacional de Estadística e Informática (INEI). [Consultado 08 de Febrero de 2011]. Disponible en Internet: <URL:<http://www.inei.gob.pe/biblioineipub/bancopub/inf/lib5047/INDEX.htm>>

³³ Ibíd., Realidad Virtual: Tipos de Sistemas Virtuales.

mundos a través de ellos, y precisamente estos son el principal elemento que lo hacen sentirse inmerso dentro de estos mundos. Se utiliza para obtener experiencias virtuales individuales. Las áreas de aplicación que emplean este sistema corresponden a las de entrenamiento, investigación, capacitación y entretenimiento³⁴.

- **Sistemas semi-inmersivos o inmersivos de proyección.** Se caracterizan por ser cuatro pantallas en forma de cubo (tres pantallas forman las paredes y una el piso), las cuales rodean al observador, el usuario usa lentes y un dispositivo de seguimiento de movimientos de la cabeza, de esta manera al moverse el usuario las proyecciones perspectivas son calculadas por el motor de Realidad Virtual para cada pared y se despliegan en proyectores que están conectados a el computador. Este tipo de sistemas son usados principalmente para visualizaciones donde se requiere que el usuario se mantenga en contacto con elementos del mundo real³⁵.

- **Sistemas no inmersivos o de escritorio.** Son aquellos donde el monitor es la ventana hacia el mundo virtual y la interacción es por medio del teclado, micrófono, mouse o joystick, este tipo de sistemas son idóneos para visualizaciones científicas, también son usadas como medio de entretenimiento y aunque no ofrecen una total inmersión son una buena alternativa como opción de bajo costo³⁶ para áreas como arquitectura, diseño, capacitación, ventas, educación y ocio, permitiendo la participación de varios usuarios conectados en red³⁷.

- **Sistemas de Simulación.** Es utilizado para simular situaciones especiales que sirven para el aprendizaje o el entrenamiento con vehículos, aviones, barcos, etcétera. Se utiliza principalmente en medios militares y aeronáuticos, aunque también se encuentran simuladores de este tipo en centros de diversión. Puede utilizarse individualmente o formando parte de una red interconectada. Suele ser usado para plataformas hidráulicas simulando sus movimientos³⁸.

En tabla 1 se mencionan los elementos utilizados en los cuatro tipos más sobresalientes de los Sistemas de Realidad Virtual (SRV).

³⁴ CORRADO PADILLA, Ericka; DELGADO, Julián; CASTAÑEDA, Salvador. Tecnologías de la Realidad Virtual: Tipos de Realidad Virtual [en línea]. [Consultado 08 de Febrero de 2011]. Disponible en Internet: <URL:<http://telematica.cicese.mx/computo/super/cicese2000/realvirtual/>>

³⁵ Ibíd., Tecnologías de la Realidad Virtual: Tipos de Realidad Virtual.

³⁶ Ibíd., Tecnologías de la Realidad Virtual: Tipos de Realidad Virtual.

³⁷ MURILLO ALFARO, Félix. Realidad Virtual: Tipos de Sistemas Virtuales [en línea]. Lima, Perú: Instituto Nacional de Estadística e Informática (INEI). [Consultado 08 de Febrero de 2011]. Disponible en Internet:<URL:<http://www.inei.gob.pe/biblioineipub/bancopub/inf/lib5047/INDEX.htm>>

³⁸ Ibíd., Realidad Virtual: Tipos de Sistemas Virtuales.

Tabla 1. Elementos utilizados en los cuatro tipos más importantes de realidad virtual.

	Dispositivos de Entrada	Unidad Central	Dispositivos de Salida
SRV de escritorio	Mouse, Palanca de mando, Controlador 3D	PC compatible, Macintosh, Estación de trabajo	Monitor de alta resolución, Gafas 3D
SRV de inmersión	Palanca de mando, Sensores de posición	PC compatible, Estación de trabajo	Casco virtual
SRV de proyección	Mouse, Palanca de mando, Controlador 3D	PC compatible, Macintosh, Estación de trabajo	Proyector de alta resolución, Pantalla
SRV de simulación	Volante, Palancas, Botones, Mandos, Interruptores	Estación de trabajo	Monitores

Fuente: MURILLO ALFARO, Félix. Realidad Virtual: Tipos de Sistemas Virtuales [en línea]. Lima, Perú: Instituto Nacional de Estadística e Informática (INEI). [Consultado 08 de Febrero de 2011]. Disponible en Internet:<URL:<http://www.inei.gob.pe/biblioineipub/bancopub/inf/lib5047/INDEX.htm>>

4.4 PLATAFORMAS PARA EL DESARROLLO DE ENTORNOS VIRTUALES

Los subsistemas gráficos necesitan de un estándar para que los programas gráficos se comuniquen con el hardware capaz de soportar gráficos 3D. A continuación se describen las plataformas más sobresalientes para el desarrollo de Entornos virtuales.

4.4.1 APIs. Son gráficos de bajo nivel (capa de abstracción del hardware). Aquí se establecen protocolos o lenguajes para enviar órdenes al motor gráfico implementado en hardware. Los APIs envían estados de *render* (transformaciones, materiales, texturas, etc.) al sistema gráfico³⁹. También se les conoce como un conjunto de técnicas, funciones y métodos utilizados para establecer una comunicación entre los diferentes niveles o capas de un software. Una API permite una abstracción en la programación entre las

³⁹ TAIBO, Javier. El Sistema Gráfico [en línea]. Coruña: España, Universidad de la Coruña, 2009. [Consultado 02 de Febrero de 2011]. Disponible en Internet: <URL:http://videalab.udc.es/i3d/docs/clases/2009_2010/Sistema_Grafico.pdf>

capas inferiores (bajo nivel) y las superiores (alto nivel), de tal forma que se pueda interactuar casi directamente con el hardware a través de funciones nativas del sistema operativo o mediante una aplicación específicamente diseñada con ese propósito⁴⁰.

- **Funciones de una API.** Una API permite a los programadores acceder a la funcionalidad de módulos de software pre-construidos. Una API define las estructuras de datos y llamadas a subrutinas. Puede proporcionar un cierto nivel de abstracción para el programador es decir, que oculta la complejidad de acceso a un sistema o aplicación, proponiendo un conjunto de funciones estándar que sólo los parámetros y valores devueltos son conocidos. Por ejemplo, analogía con un automóvil, el conductor no conoce la mecánica de operación de un vehículo de motor pero tiene el fin de conducirlo. Sólo una interfaz que consiste en un volante, los pedales (acelerador, embrague, frenos), palancas de mando (luces parpadeantes, caja de cambios) y los botones (pito, anti-niebla, alarma) sean accesibles a él: es un sentido de la interfaz de propuestas para el usuario⁴¹.

- **Características.** La propia API es un trozo de código de software escrito como una serie de mensajes XML. Cada mensaje XML corresponde a una función diferente de la de servicio remoto. Por ejemplo, en una conferencia de la API, hay mensajes XML que corresponden a cada elemento necesario para programar una conferencia Web. Esos elementos incluyen el tiempo de la conferencia, el nombre del organizador y la información de contacto, que ha invitado a la conferencia, la duración de la conferencia, etc. Los API de servicios Web son totalmente invisibles para los navegantes del sitio web y los usuarios de software. Su trabajo es correr silenciosamente en segundo plano y proporciona un camino a las aplicaciones para trabajar con otros para conseguir que el usuario acceda a la información o funcionalidad que necesita. Junto con XML, las siguientes normas tecnológicas, protocolos y lenguajes de programación son los que hacen el trabajo de servicios Web⁴².

SOAP (Simple Object Access Protocol). SOAP es responsable de los mensajes XML de codificación para que puedan ser recibidas y entendidas por cualquier sistema operativo sobre cualquier tipo de protocolo de red.

⁴⁰ PALAZZESI, Ariel; REGGIANI, Federico; ORTIZ, Kir; PARDO, Lisandro; SACCO, Mario; FERZZOLA, Max; VARONAS, Nico; GARCÍA, Tomás. DirectX y OpenGL: Las dos API más importantes del mercado gráfico [en línea]. 2006. [Consultado 10 de Febrero de 2011]. Disponible en Internet: <URL:<http://www.neoteo.com/directx-y-opengl-las-dos-api-mas-importantes-del>>

⁴¹ CREATIVE COMMONS. Lenguajes de programación- API [en línea]. España, 2008. [Consultado 10 de Febrero de 2011]. Disponible en Internet:<URL:<http://es.kioskea.net/contents/langages/api.php3>>

⁴² ROOS, Dave. ¿How APIs Work? [en línea]. Estados Unidos, 2000. How Stuff Works? Discovery Company. [Consultado 09 de Febrero de 2011]. Disponible en Internet: <URL:<http://communication.howstuffworks.com/how-to-leverage-an-api-for-conferencing2.htm>>

UDDI (Universal Description, Discovery and Integration). Descrito como un "páginas amarillas de Internet", UDDI es un directorio basado en XML que permite a las empresas a la lista, se encuentran y colaboran con los servicios Web.

WSDL (Web Services Description Language). es el SOAP WSDL del UDDI. Básicamente, WSDL es el lenguaje basado en XML que las empresas utilizan para describir sus servicios en el UDDI.

- **Ejemplos de APIs.** Los diseñadores de software escogen la API que mejor se adapte a los requerimientos del producto. Tanto Direct3D como OpenGL son APIs que se valen de librerías para cumplir con su objetivo, se describen a continuación:

OpenGL. La responsable de Open Graphics Library es Silicon Graphics Incorporated (SGI), la misma que en la década de los 90' fuera la propietaria de la API IrisGL, la antecesora de OpenGL⁴³. OpenGL es una especificación estándar abierto que define una API multilenguaje y multiplataforma (puede utilizarse en Linux, Unix, Mac OS, Windows e incluso en móviles) para escribir aplicaciones que produzcan gráficos 2D y 3D. La interfaz tiene diferentes funciones que pueden usarse para dibujar escenas tridimensionales complejas a partir de geometrías primitivas simples tales como puntos, líneas y triángulos. Se usa ampliamente en CAD, realidad virtual, representación científica, visualización de información, simulación de vuelo y en desarrollo de videojuegos.⁴⁴

Direct3D. Direct3D es una API propiedad de Microsoft. Muy extendido en el mundo de los videojuegos. Forma parte de las librerías DirectX. La API Microsoft Direct3D proporciona una interfaz para las funciones de representación en 3D incorporado en la mayoría de los adaptadores de vídeo. Direct3D es una API de bajo nivel en 3D que proporciona una capa independiente del dispositivo para que los programas de software se comuniquen con el hardware acelerador de forma eficiente y poderosa⁴⁵.

⁴³ PALAZZESI, Ariel; REGGIANI, Federico; ORTIZ, Kir; PARDO, Lisandro; SACCO, Mario; FERZZOLA, Max; VARONAS, Nico; GARCÍA, Tomás. DirectX y OpenGL: Las dos API más importantes del mercado gráfico [en línea]. 2006. [Consultado 10 de Febrero de 2011]. Disponible en Internet: <URL:<http://www.neoteo.com/directx-y-opengl-las-dos-api-mas-importantes-del>>

⁴⁴ S.A.B.I.A. Sistemas Adaptativos y Bioinspirados en Inteligencia Artificial. Open GL [en línea]. Coruña, España. [Consultado 12 de Febrero de 2011]. Disponible en internet: <URL:<http://sabia.tic.udc.es/gc/Contenidos%20adicionales/trabajos/Juegos/videojuegos-1995-2004/opengl.html>>

⁴⁵ SAQIB, Muhammad. Microsoft Direct-X Programming [en línea]. 2008. [Consultado 10 de Febrero de 2011]. Disponible en Internet: <URL:<http://www.mycplus.com/tutorials/microsoft-direct-x-programming/>>

Direct3D incluye soporte para diferentes conjuntos de instrucciones especializadas de la CPU, proporcionando una aceleración adicional en los equipos más nuevos⁴⁶.

4.4.2 Motor de juegos. Un motor de juegos debe brindar el soporte y tener la capacidad de dar una solución específica para la manipulación de elementos como datos, textos, sonidos, gráficos, entradas, salidas, etc.

- **¿Qué es un Motor de juegos (Game Engine)?**. Un motor de juegos es el núcleo software de un videojuego o de una aplicación interactiva con gráficos en tiempo real. La utilización de Motores de Juegos permite abstraer al desarrollador de muchos problemas simplificándolos en gran medida, permite desarrollar juegos multiplataforma fácilmente⁴⁷.

- **Características.** La arquitectura del motor de juegos permite la portabilidad en varias plataformas, incluyendo sistemas operativos de computadoras de escritorios, consolas de videojuegos y dispositivos móviles. A partir de un comando de un motor de juegos se pueden realizar rutinas que normalmente llevarían varios comandos en una librería gráfica como OpenGL o DirectX. El motor gráfico o de render proporciona funciones de renderizado 2D, 3D y de sprites. El motor gráfico se encarga de la visibilidad, el mapeo de texturas, antialiasing, la gestión de mallas en 3D, entre otras cosas. El grafo de escena se encarga de ordenar la representación lógica y espacial de una escena. Una escena se representa con una estructura de datos en forma de árbol, en donde el nodo raíz es el escenario principal. Los nodos hijos son los objetos que están en el escenario y que a su vez constan de varios hijos. El motor de física simula los modelos del mundo utilizando variables como la velocidad, masa, gravedad, fuerza, etc. Este tipo de simulaciones hace ver más real las acciones del mundo virtual, creando un mejor impacto visual⁴⁸.

- **Arquitectura de un motor.** El diseño del motor se centra en la flexibilidad, teniendo en cuenta la simple expansión de su funcionalidad. Dicho esto, se puede modificar fácilmente para adaptarse a las plataformas que se ven limitados por

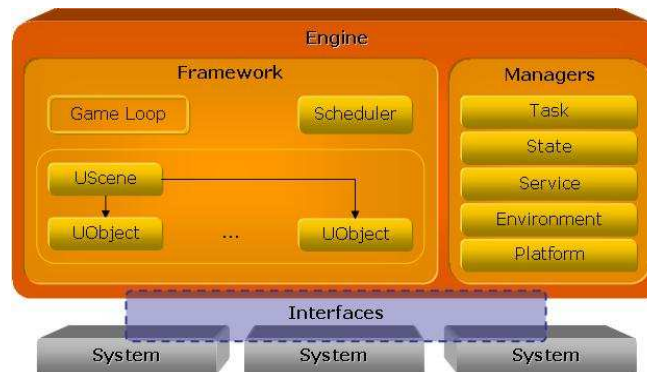
⁴⁶ What is DirectX? [en línea]. E.U.: Microsoft Corporation, 2011. [Consultado 10 de Febrero de 2011]. Disponible en Internet: <URL:http://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/what_is_directx.mspx?mfr=true>

⁴⁷ CORTIZO PÉREZ, José Carlos. Motores de Juegos [en línea]. Madrid: Universidad Europea de Madrid, Máster en Diseño y Programación de Videojuegos. [Consultado 10 de Febrero de 2011]. Disponible en internet: <URL:<http://www.esi.uem.es/jccortizo/motores.html>>

⁴⁸ MADERA RAMÍREZ, Francisco A. Herramientas de Programación Gráfica para Desarrollo de Videojuegos [en línea]. México, D.F.: Universidad Autónoma de Yucatán. [Consultado 14 de Febrero de 2011]. Disponible en Internet: <URL:<https://intranet.matematicas.uady.mx/personal/mramirez/website/madera/unacarPaper.pdf>>

ciertos factores, como la memoria, etc. El *framework* contiene las partes del juego que se duplican, es decir, habrá varias instancias de ellos. También contiene elementos que tienen que ver con la ejecución del bucle principal del juego. Los administradores son hijos únicos que dependen de la lógica del juego⁴⁹. La Figura 2 ilustra las diferentes secciones que componen el motor de alto nivel.

Figura 2. Arquitectura de un motor de alto nivel



Fuente: intel ® Software Network. Designing the Framework of a Parallel Game Engine [en línea]. [Consultado 12 de febrero de 2011]. Disponible en Internet: <URL:<http://software.intel.com/en-us/articles/designing-the-framework-of-a-parallel-game-engine/>>

Con el propósito de modularidad, el motor y la funcionalidad de procesamiento de juego o sistema, son entidades que deben permanecer separadas. Las interfaces son los medios de comunicación entre el motor y los sistemas. Los sistemas implementan las interfaces para que el motor tenga acceso a la funcionalidad de un sistema, y el motor implementa la interfaz para que los sistemas puedan acceder al módulo de administración.⁵⁰

- **Ejemplos.** Los motores de juego modernos, proporcionan un escenario gráfico, que es una representación orientada a objetos del mundo del juego en 3D, que a menudo se encarga de simplificar el diseño de los juegos y se puede utilizar para obtener una prestación más eficiente de los infinitos mundos virtuales. Algunos ejemplos de los motores gráficos son: RealmForge, Blender, Truevision3D, OGRE, Crystal Space, Genesis3D, Vision Engine, motor de Irrlicht Panda 3D y jMonkey. A continuación dos de los motores de juegos modernos y de código abierto que se han hecho reconocer gracias a su portabilidad y facilidad de uso.

⁴⁹ Intel® Software Network. Designing the Framework of a Parallel Game Engine [en línea]. [Consultado 12 de Febrero de 2011]. Disponible en internet: <URL:<http://software.intel.com/en-us/articles/designing-the-framework-of-a-parallel-game-engine/>>

⁵⁰ Ibíd., Designing the Framework of a Parallel Game Engine.

jMonkey Engine. Es un motor 3D de código abierto. Está programado completamente en Java y compatible con OpenGL, hecho especialmente para los desarrolladores que quieren crear juegos en 3D con las normas y estándares de la tecnología moderna y que están destinados a la amplia accesibilidad.

Su característica principal es que es multiplataforma, ya que al funcionar en Java es posible usarlo desde aplicaciones web hasta aplicaciones para teléfonos móviles. La apariencia de este software es impresionante, además de que también cuenta con un motor físico, lo cual resulta muy útil a la hora de crear juegos.

jMonkeyEngine puede ser un punto de partida para cualquier desarrollador aspirante a los juegos 3D, no sólo los programadores de Java, sino cualquier persona capaz de adaptarse a la programación Java. Si bien el alcance de los juegos 3D y la tecnología es enorme, jMonkeyEngine envuelve todos los fundamentos en un paquete pequeño y atractivo para los desarrolladores para ser explorado paso a paso. Aún así, el verdadero poder de jME sólo está limitado por el poder de OpenGL y la imaginación.⁵¹

Panda 3D. Es un motor de juegos, una plataforma diseñada para la representación y desarrollo de juegos usando programas como Python y C++. Panda 3D es de código abierto y libre para cualquier propósito, incluyendo empresas comerciales. Panda 3D fue desarrollado por Disney y por el Centro de Tecnología de Carnegie Mellon University.

Panda 3D expone características modernas entre OpenGL y DirectX a los usuarios con altos niveles de sombreado, interfaces de gran alcance entre los sombreados y el motor, tiene soporte para representar gráficos sobre texturas. Su sistema es usado para entregar diferentes juegos comerciales como los importadores de código fuente, convierte y empaqueta los juegos haciéndolos redistribuibles.⁵²

- **Motores en la actualidad.** Como la tecnología del motor del juego madura y se vuelve más fácil de usar, la aplicación de motores de juego se ha ampliado al alcance de todo quien quiera usarlo, y ahora están siendo utilizados para la visualización de mundos virtuales con mayor seriedad y compromiso, como: la formación de médicos y las aplicaciones militares de simulación.

Para facilitar esta accesibilidad, nuevas plataformas de hardware están siendo dirigidas por los motores de juego, incluyendo los teléfonos (por ejemplo, el iPhone

⁵¹ jME Wiki. What is jMonkeyEngine? [En línea]. [Consultado: 15 de Febrero, 2011]. Disponible en internet: <URL:<http://jmonkeyengine.org/introduction/>>

⁵² Panda 3D. Basics and features. [En línea]. [Consultado: 05 de Abril, 2011]. Disponible en <URL:internet: <http://www.panda3d.org>>

y Android) y navegadores Web (por ejemplo, Shockwave, Flash, Trinigy's Webvision, Silverlight, Unity Web Player, O3D y DHTML puro)⁵³.

Cada vez más los motores de juego se basan en lenguajes de alto nivel tales como Java y C# / .NET (por ejemplo, TorqueX y Visual3D.NET) o Python (Panda3D). Como la mayoría de los juegos 3D se ven afectados por la GPU limitada (es decir, limitada por la potencia de la tarjeta gráfica), la ralentización del potencial de lenguajes de alto nivel tiende a ser insignificante, mientras que el aumento de la productividad que ofrecen estos lenguajes trabajan en beneficio de los desarrolladores del motor de juego. Estas recientes tendencias están siendo impulsadas por empresas como Microsoft para apoyar el desarrollo independiente de juegos. Microsoft desarrolló XNA como el SDK de elección para todos los videojuegos lanzado en Xbox y otros productos relacionados.⁵⁴ Esto incluye el Xbox Live Community Games, canal diseñado específicamente para los pequeños desarrolladores que no cuentan con los amplios recursos necesarios a los juegos de caja para la venta en las tiendas⁵⁵.

4.4.3 Librerías gráficas. Se pueden definir como el conjunto de funciones y clases que generan representaciones de imágenes. Utilizan para ello modelos matemáticos junto con patrones de texturas, iluminación, colores, etc.

Se requiere tener un buen conocimiento del hardware de representación de datos 2D/3D, para conocer las posibilidades y limitaciones, haciendo un estudio cuidadoso de todos los procesos que se deben llevar a cabo antes de llegar a ver en pantalla el objeto 3D⁵⁶.

- **Librerías de bajo nivel.** Las librerías de bajo nivel están orientadas a facilitar la utilización hardware generador de gráficos, pero resultan insuficientes si se desea gestionar una escena compleja, y casi todas las escenas de realidad virtual lo son. Uno de los objetivos principales de las librerías de bajo nivel es proporcionar un API que sea independiente del hardware gráfico, y que al mismo tiempo, permita un control total sobre sus funcionalidades (p.ej. OpenGL) y proporcione acceso a operaciones gráficas del nivel más bajo posible, dejando en

⁵³ BORICK, Stephen. Game Engine Trends [en línea]. JBSA Technologies, Innovate Technologies in Multimedia, Graphics and 3D Design, 2009. [Consultado 14 de Febrero de 2011]. Disponible en Internet:<URL:http://jbsatech.com/index.php?option=com_content&view=section&layout=blog&id=3&Itemid=6>

⁵⁴ Ibíd.,. Game Engine Trends.

⁵⁵ Carles, S. 2011 UBM TechWeb. BREAKING: Microsoft To Enable User-Created Xbox 360 Games. [En línea]. [Consultado: 15 de Febrero, 2011]. Disponible en internet: <URL:http://www.gamasutra.com/php-bin/news_index.php?story=10458>

⁵⁶ SERRANO BÁRCENA, Nicolás; MATEY, Luis. Librerías Gráficas: Objetivos de las librerías gráficas [en línea]. España: Universidad de Navarra, 2010-2011. [Consultado 15 de Febrero de 2011]. Disponible en Internet:<URL:<http://www.tecnun.es/asignaturas/grafcomp/presentaciones/pdf/libreriasGraficas6.pdf>>

manos del desarrollador de la aplicación, la creación y gestión de objetos geométricos más complejos tales como cubos, esferas, autos, casas o animales. Las librerías de bajo nivel proporcionan métodos para realizar únicamente descripciones geométricas muy sencillas (por ejemplo un triángulo con un material y una textura determinada)⁵⁷.

- **Librerías de alto nivel.** También llamadas librerías descriptivas, estas permiten desarrollos de aplicaciones 3D más rápidos que las librerías de rendering directo. En este caso, la programación se simplifica más debido a que se focaliza cada representación de las imágenes mediante un árbol de escena.

Las características más sobresalientes de esta librería son que se encarga de gestionar la carga/descarga de memoria, el uso del buffer gráfico y los elementos no visibles de la escena (evitando renderizarlos, para ahorrar recursos del sistema y tiempo de procesado) etc.

Existen librerías de más alto nivel que pueden ayudar al programador en la tediosa tarea de crear y gestionar objetos geométricos, encapsulando la complejidad de estos objetos en estructuras especiales, y además, proporcionando métodos para gestionar la inmensa cantidad de información geométrica que suele ser manejada. La primera necesidad de un desarrollador de aplicaciones de realidad virtual es por ejemplo, crear un triángulo con un material y una textura determinada, luego agrupar ese triángulo con otros similares para poder definir la geometría de un tornillo. A su vez, agrupar este tornillo con otra serie de elementos, para formar otro objeto de mayor complejidad, como puede ser una lámpara; agrupar esa lámpara con otros objetos para formar una habitación, etc⁵⁸.

Esta forma en la que los objetos del mundo real aparecen agrupados de forma jerárquica, ya podría ser una razón suficiente para que las librerías de alto nivel tomen la decisión de emplear una estructura de datos con una organización jerárquica. Además, la organización jerárquica de los elementos que constituyen una escena, facilita la utilización de múltiples sistemas de referencia, y ayuda a controlar la carga del sistema. La estructura jerárquica empleada de forma generalizada por todas las librerías gráficas de alto nivel, recibe el nombre de Grafo de Escena o *Scene Graph* en literatura inglesa⁵⁹.

⁵⁷ MARTÍNEZ RODILLA, Miguel. Desarrollo de un sistema de edición de escenas basado en una librería gráfica de alto nivel [en línea]. Valencia, España, 2006. Trabajo de grado (Ingeniero en Informática). Universidad de Valencia. [Consultado 03 de Marzo de 2011]. Disponible en Internet: <URL:http://mural.uv.es/mimaro/proyecto_final.pdf> p.24.

⁵⁸ Ibíd.,. Desarrollo de un sistema de edición de escenas basado en una librería gráfica de alto nivel, p.24.

⁵⁹ MARTÍNEZ RODILLA, Miguel. Desarrollo de un sistema de edición de escenas basado en una librería gráfica de alto nivel [en línea]. Valencia, España, 2006. Trabajo de grado (Ingeniero en Informática). Universidad de Valencia. [Consultado 03 de Marzo de 2011]. Disponible en Internet: <URL:http://mural.uv.es/mimaro/proyecto_final.pdf> p. 24.

4.4.4 Grafo de Escena (Scene Graph). Un grafo de escena es un grafo dirigido acíclico de nodos que contiene los datos que definen un escenario virtual y controlan su proceso de dibujado. Contiene descripciones de bajo nivel de la geometría y la apariencia visual de los objetos, así como descripciones de alto nivel referentes a la organización espacial de la escena, datos específicos de la aplicación, transformaciones, etc. Los grafos de escena almacenan la información del escenario virtual en diferentes tipos de nodos.

Existen nodos que almacenan la información geométrica y actúan como nodos hijos dentro del grafo de escena; el resto de los nodos suelen aplicar algún tipo de modificación sobre el segmento de jerarquía que depende de ellos, bien sea estableciendo agrupaciones, aplicando alguna transformación similar o realizando algún tipo de selección sobre alguna de sus ramas hijas. El proceso de dibujado consiste en realizar un recorrido de dicho grafo, aplicando las operaciones indicadas por cada tipo de nodo⁶⁰.

- **Funciones principales de un grafo de escena.** Un grafo de escena principalmente contribuye a establecer una organización lógica de la escena⁶¹; establece dependencias jerárquicas entre distintos sistemas de referencia; posibilita el proceso de selección entre múltiples niveles de detalle; posibilitar el proceso automático de *Culling*⁶². Facilita el control de la escena por parte del usuario y hace más cómodo el acceso a las librerías gráficas de bajo nivel⁶³.

4.5 MODELADO Y TEXTURIZADO DE OBJETOS 3D

Durante la elaboración de escenas 3D, el manejo de las luces es de vital importancia, tan vital como el adecuado control de las sombras proyectadas. Las sombras son un factor clave para el realismo, donde se debe considerar también la creación de materiales con su opacidad y reflejos.

4.5.1. ¿Qué es modelar? Se trata de un proceso de creación de un modelo de estructura geométrica que representa un objeto tridimensional. Ese objeto puede estar *vivo* o *inanimado*. Un modelo tridimensional se crea mediante un conjunto de puntos en el espacio 3D, que están conectadas por varios datos geométricos,

⁶⁰ Ibíd., Desarrollo de un sistema de edición de escenas basado en una librería gráfica de alto nivel, p.24, 25.

⁶¹ Es una estructura de datos que describe los escenarios y los objetos que habitan ese espacio como geometrías, luces, cámaras, apariencia (colores, materiales, texturas, etc.)

⁶² Eliminación automática de los objetos que se encuentran fuera del campo de visión.

⁶³ Ibíd., Desarrollo de un sistema de edición de escenas basado en una librería gráfica de alto nivel, p. 25.

como líneas y superficies curvas. Modelado en 3D es el procedimiento de elaboración de un modelo en 3D mediante software especializado⁶⁴.

Hoy en día, el modelado 3D se emplea en una amplia variedad de actividades. La industria de los videojuegos cuenta con tres métodos diferentes de modelado tridimensional para crear modelos realistas de personajes. En el campo de la medicina, modelos detallados de los órganos del cuerpo humano son creados usando una serie de software avanzado de modelado 3D. La comunidad de ingenieros usan los programas de diseño asistido por computador (Computer Aided Design, CAD) en 3D para crear modelos tridimensionales de los nuevos dispositivos, vehículos y estructuras.⁶⁵

4.5.2. ¿Qué es texturizar? Es un método para añadir cualidades de superficies como la madera, plástico, metal; a un modelo 3D. Esta aplicación a los modelos 3D fue creada por Edwin Catmull que publicó los resultados de su tesis en 1974⁶⁶, técnica denominada mapeo de textura o “texture mapping” en literatura inglesa. Texture mapping es la acción de aplicar texturas 2D a modelos 3D. Estas texturas se aplican al modelo como un envoltorio externo que consigue proporcionar realismo en los modelos sin aumentar el número de polígonos⁶⁷.

4.5.3. ¿Qué es un modelo 3D? Un modelo 3D se desarrolla en este espacio con reglas de la geometría medida por las coordenadas cartesianas X, Y, Z para representar a la anchura, altura y profundidad⁶⁸. Los siguientes son los métodos más populares para crear modelos tridimensionales:

Modelado poligonal, muchos modelos tridimensionales se crean como textura modelos poligonales. Modelado poligonal es un método de crear un modelo en 3D mediante la conexión de segmentos de línea a través de puntos en un espacio 3D. Estos puntos en el espacio también se conocen como vértices. Modelos

⁶⁴ GHOSH, Anurag. What is 3D Modeling? [en línea]. Conjecture Corporation, 2003-2011. [Consultado 17 de Febrero de 2011]. Disponible en internet: <URL:<http://www.wisegeek.com/what-is-3d-modeling.htm>>

⁶⁵ Ibíd., What is 3D Modeling?

⁶⁶ ROGEZ, Benoit. Histoire de la 3D [en línea]. Francia, 2007-2009. [Consultado 03 de Marzo de 2011]. Disponible en Internet: <URL:<http://www.shadows.fr/3d/chronologie?page=1>>

⁶⁷ Escuela de animación: SKALA VFX. Texture Mapping [en línea]. México, Heroica Puebla de Zaragoza. [Consultado 04 de Marzo de 2011]. Disponible en Internet: <URL:<http://www.skalavfx.com/TextureMapping.html>>

⁶⁸ Ibíd., What is 3D Modeling?

poligonales son muy flexibles, sin embargo, no es posible crear una superficie curva exacta utilizando la técnica de modelado poligonal 3D⁶⁹.

Modelos primitivos, esta es la forma más sencilla de modelar objetos tridimensionales. Uso de primitivas geométricas tales como cilindros, conos, cubos y pelotas, se pueden crear modelos complejos. Este enfoque garantiza la construcción fácil ya que las formas son matemáticamente definidas y precisas. Modelos primitivos se utilizan principalmente en el desarrollo de modelos 3D de aplicaciones técnicas⁷⁰.

Modelado NURBS, método de modelado 3D, se puede encontrar en el software popular Maya. El desarrollador puede crear modelos en 3D con superficies lisas con efectos suaves, a diferencia de las técnicas de modelado poligonal que sólo puede aproximarse a superficies curvas con numerosos polígonos⁷¹.

4.5.4. Finalidad del modelado y texturizado de objetos 3D. La finalidad de un modelo 3D es ser representado en un espacio virtual que sea parecido a lo que se conoce en nuestra realidad; la finalidad de la textura es hacer que ese modelo 3D pueda ser representado como un objeto real: que se conecte con un objeto de nuestra memoria logrando el efecto de ilusión.

4.5.5. Herramientas de modelado y texturizado de objetos

Hay muchas aplicaciones de modelado 3D disponibles en el mercado. El extremo más alto de este mercado produce programas muy complejos que puede llevar años de estudio para dominar por completo. El otro software de modelado 3D consiste en sólo el conjunto de herramientas básicas necesarias para crear modelos 3D, que en general son más fáciles de aprender.⁷²

- **Wings 3D.** Es un modelador de objetos 3D, originalmente inspirado en Nendo y Mirai de Izware, Wings se ha desarrollado desde 2001, cuando Björn Gustavsson y Gudmundsson Dan iniciaron el proyecto. Ofrece una amplia gama de herramientas de modelado, una interfaz personalizable, soporte de luces y materiales, y un constructor en las instalaciones de la cartografía de texturas AutoUV. Wings no tiene soporte para la animación.

⁶⁹ GHOSH, Anurag. What is 3D Modeling? [en línea]. Conjecture Corporation, 2003-2011. [Consultado 17 de Febrero de 2011]. Disponible en internet: <URL:<http://www.wisegeek.com/what-is-3d-modeling.htm>>

⁷⁰ Ibíd., What is 3D Modeling?

⁷¹ Ibíd., What is 3D Modeling?

⁷² MCGREGOR, Nicolas ¿What is Polygon Modeling?: overview of How to Model in 3D Modeling Software Applications [en línea]. Suite101, 2008. [Consultado 18 de Febrero de 2011]. Disponible en Internet: <URL:<http://www.suite101.com/content/what-is-3d-polygon-modeling-a73217>>

Wings 3D fue desarrollado en Erlang, lenguaje de programación de código abierto funcional distribuido por Ericsson y totalmente gratuito para uso en proyectos personales y comerciales⁷³.

- **Blender 3D.** Blender es software de código abierto; es una suite de creación de contenido 3D, disponible para los principales sistemas operativos bajo la General Public License (GNU). Blender tiene una estructura estrictamente orientada a objetos. Cada aspecto del mundo 3D ha sido organizado en pequeños bloques de datos. Realizando enlaces entre ellos, haciendo copias y editando o re-usándolos, se pueden construir entornos complejos con el uso mínimo de memoria.⁷⁴

Plataformas compatibles con Blender: Windows 2000, XP, Vista; Mac OS X (PPC e Intel); Linux (i386), Linux (PPC); FreeBSD 5.4 (i386); SGI Irix 6.5; Sun Solaris 2.8 (SPARC)

- **Maya.** Es un software de animación que ofrece un flujo de trabajo creativo de extremo a extremo con herramientas completas para la animación, modelado, simulación, efectos visuales, la representación de movimientos, y de composición en una plataforma de producción altamente extensible.

Maya está diseñado para ayudar a satisfacer las necesidades más exigentes en la producción de modelos, escenas, entornos 3D para cine, juegos, televisión, publicidad, editorial o diseño gráfico. Autodesk® Maya® 2012 ofrece un conjunto de herramientas nuevas para la pre-visualización y la creación de un prototipo de juego con capacidades ampliadas de simulación. La ventana gráfica de alto rendimiento introducidas con Maya 2011 ahora ofrece efectos visuales a pantalla completa: el desenfoque de movimiento, la profundidad de campo y oclusión ambiental, lo que le permite evaluar el trabajo realizado en un entorno de mayor fidelidad y sin necesidad de hacer exportación a un motor de juego.⁷⁵

- **3ds Max.** Autodesk 3ds Max ofrece un modelo completo e integrado en 3D, se presenta como una solución para los desarrolladores de juegos, artistas de efectos visuales y diseñadores gráficos que deseen los efectos animación y renderizado a sus modelos 3D.

Contiene una potente herramienta para el modelado 3D poligonal extenso y un conjunto de herramientas de mapeado de texturas, modeladores y textura artística para trabajar de forma más rápida y eficaz.

Usa una manipulación avanzada de animación de personajes totalmente integrado. El Kit de herramientas de animación de caracteres (CAT) proporciona

⁷³ ¿What is Wings 3D? [en línea]. Wings 3D, 2011. [Consultado 18 de Febrero de 2011]. Disponible en Internet: <URL:<http://www.wings3d.com/>>

⁷⁴ Blender Manual [en línea]. Blender is the free open source 3D content creation suite, 2011. [Consultado 18 de Febrero de 2011]. Disponible en internet: <URL:<http://www.blender.org>>

⁷⁵ Product Information [en línea]. Autodesk, Inc. Autodesk Maya, 2011. [Consultado 22 de Febrero de 2011]. Disponible en internet: <URL:<http://usa.autodesk.com/maya/>>

funcionalidades avanzadas de equipamiento y sistema de animación. Posee una extensible canalización de apoyo a desarrolladores para que puedan personalizar, ampliar e integrar 3ds Max con soporte para C + + y .NET.⁷⁶

- **Rhinoceros 3D.** Proporciona las herramientas para modelar con precisión y documentar los diseños listos para el renderizado, animación, el dibujo, la ingeniería, el análisis y la fabricación o construcción.

Rhino puede crear, editar, analizar, documentar, renderizar, animar y traducir curvas NURBS, superficies y sólidos sin límite de complejidad, grado o tamaño. Rhino también es compatible con mallas poligonales y mallas de puntos.

Rhino ofrece la forma libre de modelado 3D. Diseño, prototipos, ingeniería, análisis, documentación y fabricación de cualquier cosa en cualquier tamaño con alta precisión del modelo, disponiendo de una corta curva de aprendizaje, donde el usuario únicamente debe centrarse en el diseño y la visualización sin distraerse por el software.⁷⁷

⁷⁶ Product Information [en línea]. Autodesk, Inc. Autodesk 3DS max, 2011. [Consultado 22 de Febrero de 2011]. Disponible en internet: <URL:<http://usa.autodesk.com/3ds-max/>>

⁷⁷ Modeling tools for designers [en línea]. Rhinoceros, 2010. [Consultado 22 de Febrero de 2011]. Disponible en internet: <URL:<http://www.rhino3d.com/>>